



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/772,650	01/30/2001	Harm Sluiman	CA920000042US1	1018
61136	7590	07/23/2009	EXAMINER	
HAMILTON & TERRILE, LLP			KANG, INSUN	
IBM RSW				
P.O. BOX 203518			ART UNIT	PAPER NUMBER
AUSTIN, TX 78720			2193	
			NOTIFICATION DATE	DELIVERY MODE
			07/23/2009	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

tmunoz@hamiltontertile.com

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Ex parte HARM SLUIMAN

Appeal 2008-005151¹
Application 09/772,650
Technology Center 2100

Decided:² July 21, 2009

Before JEAN R. HOMERE, JOHN A. JEFFERY, and
STEPHEN C. SIU, *Administrative Patent Judges*.

HOMERE, *Administrative Patent Judge*.

DECISION ON APPEAL

¹ Filed January 30, 2001. The real party in interest is International Business Machines Corporation.

² The two-month time period for filing an appeal or commencing a civil action, as recited in 37 C.F.R. § 1.304, begins to run from the decided date shown on this page of the decision. The time period does not run from the Mail Date (paper delivery) or Notification Date (electronic delivery).

I. STATEMENT OF THE CASE

Appellant appeals under 35 U.S.C. § 134(a) from the final rejection of claims 1 through 8. We have jurisdiction under 35 U.S.C. § 6(b).

We reverse.

Appellant's Invention

Appellant invented a method and system for automatically testing a software test component. (Spec. 1, ll. 4-5.) As depicted in Figure 1, a TestCase (10) issues a test code that generates test or trace data to record its interaction with an interface of the software test Component (12) via a call and return pair (14, 16) associated with the software test Component. (Spec. 5, ll. 4-8.) Upon receiving the test code, the Component interface generates a wrapper (18) equipped with another interface having a call and return pair (20, 22) that mirrors the call and return pair (14, 16) of the Component interface to thereby receive calls from the software test component (12). (Spec. 8, ll.1-3.) Thus, upon receiving a call from the TestCase (10) via a first call (20, 22), the wrapper (18) delegates or passes it to the software test component (12) via a second call (14, 16). Conversely, returned values follow the same route in reverse. (Spec. 8, ll. 3-7.)

Illustrative Claim

Independent claim 1 further illustrates the invention. It reads as follows:

1. A method for testing a software test component, said method comprising the steps of:
ascertaining a public interface of the software test component; and
creating a wrapper component for the software test component by the substeps of:
defining a wrapper component interface to mirror the public interface of the

software test component and to receive calls to the software test component, defining the wrapper component to delegate to the software test component by including calls to the public interface of the software test component within the wrapper component, inserting test code within the wrapper component to permit capture and playback of user interaction with the public interface of the software test component, and enabling a test case to use the wrapper component interface to pass the received calls to the software test component and to generate test data from the test code in the wrapper component.

Prior Art Relied Upon

The Examiner relies on the following prior art as evidence of unpatentability:

Kobayashi US 6,633,888 B1 Oct. 14, 2003
APA (Applicant's Admitted Prior Art) disclosed in the instant application

Rejection on Appeal

The Examiner rejects the claims 1 through 8 under 35 U.S.C. § 103(a) as being unpatentable over the combination of Kobayashi and Appellant's Admitted Prior Art ("APA").

Appellant's Contentions

Appellant contends that the Examiner erred in concluding that the combination of Kobayashi and APA renders independent claim 1 unpatentable. (App. Br. 4-6.) In particular, Appellant argues that Kobayashi does not teach or suggest a test case interface mirrored on both sides of a created wrapper. (App. Br. 4.) Rather, Kobayashi teaches adapters that allow a visual editor to edit proxy beans, and to subsequently test the edited code through a universal transport API. (App. Br. 4-5.)

Examiner's Findings and Conclusions

The Examiner finds that Kobayashi's disclosure of a bean compiler that extracts member class information (e.g., properties, methods, constructors) from each component to convert it into a corresponding proxy component teaches a mirror that reflects an original component. (Ans. 6-7.)

II. ISSUE

Has Appellant shown that the Examiner erred in finding that Kobayashi teaches or suggests creating a wrapper component of a software test component having a wrapper component interface that mirrors a public interface of the software test component, as recited in independent claim 1?

III. FINDINGS OF FACT

The following Findings of Fact (FF) are shown by a preponderance of the evidence.

Kobayashi

1a. Kobayashi discloses a method and system for creating a proxy component to test a component class code and to build bean-based applications from existing object-oriented programming (OOP) object and component code using a conventional visual builder. (Abstract, ll. 1-2, col. 7, l. 66- col. 8, l. 2.)

1b. As shown in Figure 2, upon receiving objects and components (204) that are to be used with the system, a bean compiler (208) parses the text based Java class code in each of the received component to extract therefrom relevant methods and parameters to subsequently create a proxy

component (210) corresponding to each of the received components. (Col. 8, ll. 8-14, 32-38.)

1c. Upon receiving the created proxy components, a visual builder (214) manipulates them to create a bean-based application (216). Each composite object in the application can be tested using a universal transport API (206), which operates under the control of constructor and method objects instantiated by the proxy beans within the bean-based application to call the appropriate parameters for the target class in the implementation code. (Col. 8, ll. 19-27, col. 9, ll. 44-48.)

IV. PRINCIPLES OF LAW

Burden on Appeal

Appellant has the burden on appeal to the Board to demonstrate error in the Examiner's position. *See In re Kahn*, 441 F.3d 977, 985-86 (Fed. Cir. 2006) (“On appeal to the Board, an applicant can overcome a rejection by showing insufficient evidence of *prima facie* obviousness or by rebutting the *prima facie* case with evidence of secondary indicia of nonobviousness.”) (quoting *In re Rouffet*, 149 F.3d 1350, 1355 (Fed. Cir. 1998)).

V. ANALYSIS

Independent claim 1 recites, in relevant part, creating a wrapper component of a software test component having a wrapper component interface that mirrors a public interface of the software test component. As set forth in the Findings of Fact section, Kobayashi discloses a system wherein a bean compiler creates proxy components using parameters extracted from received components associated therewith. (FF. 1a, 1b.)

Further, Kobayashi discloses that the created proxy components are used to generate a bean-based application for testing each received composite object using a universal transport API. (FF. 1c.) We find that Kobayashi's disclosure teaches or suggests, at best, creating a proxy component as a delegate to test an original component associated therewith. However, we find that Kobayashi's disclosure is devoid of any teaching or suggestion that the created proxy component includes a pair of interfaces that mirror each other. We do not agree with Examiner that Kobayashi's extraction of parameters from an original component to create the proxy component teaches or suggests the recited limitation. Such a finding, in our view, is untenable and unreasonable. The extracted parameters only permit the created proxy component to have a one-way communication with the visual builder, whereas the created wrapper of Appellant's invention enables the Testcase to have a bi-directional communication with the Component. Further, we agree with Appellant that APA does not cure the noted deficiencies of Kobayashi. It follows that Appellant has shown that the Examiner erred in concluding that the combination of Kobayashi and APA renders claim 1 unpatentable.

Because claims 2 through 8 also recite these same limitations, we find that Appellant has shown error in the Examiner's rejection of those claims.

VI. CONCLUSION OF LAW

Appellant has shown that the Examiner erred in concluding that claims 1 through 8 are unpatentable as set forth above.

Appeal 2008-005151
Application 09/772,650

VII. DECISION

We reverse the Examiner's decision to reject claims 1 through 8.

REVERSED

Erc

HAMILTON & TERRILE, LLP
IBM RSW
P.O. BOX 203518
AUSTIN TX 78720